

Challenge Machine Learning : Classification en phase de sommeil avec Dreem

Marion Favre d'Echallens et Jean-Louis Truong

7 janvier 2019

1. Introduction

Ce challenge est réalisé en partenariat avec l'entreprise Dreem qui est une start-up spécialisée dans l'amélioration du sommeil des personnes.

Contexte du challenge

Ce challenge consiste à réaliser de la classification en stades de sommeil. Une nuit voit défiler plusieurs cycles de sommeil qui se composent tous d'une phase :

- d'éveil
- de sommeil léger
- de sommeil profond
- de sommeil paradoxal.

Un moyen de mesurer le sommeil est d'utiliser le polysomnographe qui relève notamment l'activité du cerveau, le mouvement des yeux et la tension musculaire afin d'évaluer la qualité du sommeil d'une personne.

Dans cette optique de mesure, la société Dreem a développé un bandeau qui fonctionne comme le polysomnographe et qui permet de mesurer trois types de signaux:

- l'activité électrique du cerveau grâce à un électro-encéphalogramme (EEG)
- le mouvement la position, la respiration grâce à un accéléromètre
- les battements sanguins grâce à un oxymètre de pouls.

Le challenge

Ce bandeau enregistre donc une certaine quantité de données par nuit et l'objectif de ce challenge est de développer un algorithme permettant, à partir des données de 30 secondes d'enregistrement du bandeau, dans quel stade de sommeil parmi les quatre cités plus haut se trouve la personne.

Nous avons pour cela à notre disposition 7 enregistrements d'encéphalogramme (sept positions différentes sur la tête), 1 enregistrement d'oxymètre et 3 enregistrements d'accéléromètre. Ces enregistrements sont de 30 secondes et ils sont labellisés i.e. nous connaissons le stade de sommeil associé.

2. Prétraitement des données

Les données sont présentées sous le format h5 afin de faciliter leur manipulation au vu de leur taille très volumineuse. Nous disposons en effet de sept bases de données d'enregistrements d'encéphalogrammes contenant chacun 38289 lignes de 1500 valeurs, ce qui correspond à une fréquence de 50Hz. Les quatre autres bases de données ne contiennent que 300 valeurs par enregistrement (fréquence de 10Hz).

Afin de lire et manipuler ces données, nous utilisons le package `h5` de R.

L'objet `xtrain` contient les onze datasets à exploiter. Pour ce faire, nous les transformons en `dataframe` afin de les utiliser.

```
library(h5,warn.conflicts = FALSE)
```

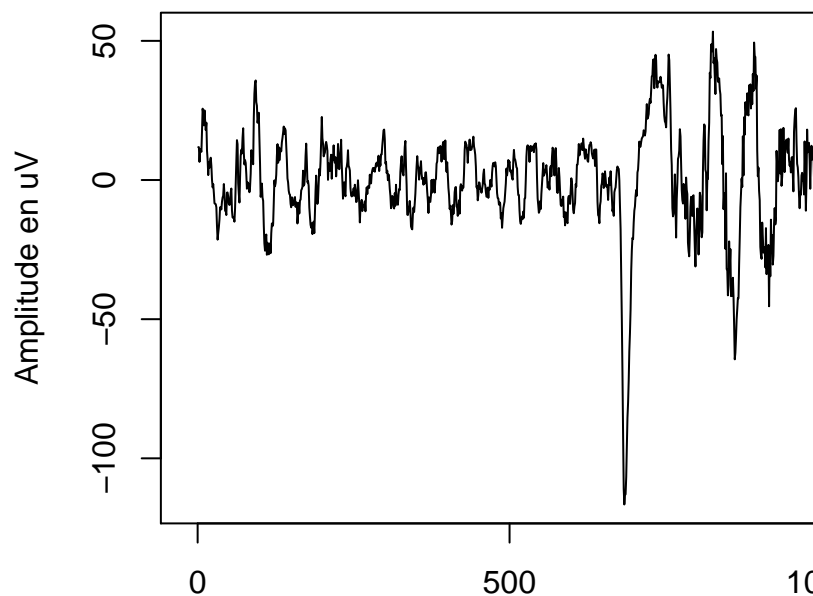
```
## Warning: package 'h5' was built under R version 3.4.4
```

```
data_folder = "C:/Users/Admin/Documents/Centrale Paris/3A/OMA/Machine Learning/Challenge/Data/"
ytrain = read.csv(paste0(data_folder,"train_y.csv"))
xtrain = h5file(name = paste0(data_folder,"train.h5/train.h5"))
list.datasets(xtrain)
```

```
## [1] "/accelerometer_x"      "/accelerometer_y"
## [3] "/accelerometer_z"      "/eeg_1"
## [5] "/eeg_2"                "/eeg_3"
## [7] "/eeg_4"                "/eeg_5"
## [9] "/eeg_6"                "/eeg_7"
## [11] "/pulse_oximeter_infrared"
```

```
eeg1 = xtrain[list.datasets(xtrain, recursive = TRUE)[4]]
eeg1 = as.data.frame(readDataSet(eeg1))
```

EEG position 1 – enreg



30 secondes d'enregistrement `une fré

On peut observer le premier enregistrement ci-dessous.

3. Extraction de features

Afin de construire un modèle de classification des données en stade de sommeil, nous avons extrait des signaux un certain nombre de features. Nous les avons ensuite testés en appliquant l'algorithme de classification présenté dans la section suivante afin de déterminer l'importance de leur influence sur la détermination du stade de sommeil.

Nous avons utilisé pour différentes approches pour le choix des features à extraire.

Nous avons d'abord calculé des features basiques sur tous les signaux: l'écart-type et la moyenne du signal, la moyenne, le minimum et le maximum du signal en valeur absolue.

Ensuite, en nous appuyant sur les ondes caractéristiques présentes dans les différents stades de sommeil nous avons réalisé deux types de décomposition des signaux EEG. En effet, on peut distinguer quatre types d'onde entre 0 et 30Hz:

- les ondes alpha entre 8 et 13Hz
- les ondes theta entre 4 et 8Hz
- les ondes beta entre 13 et 30Hz
- les ondes delta entre 0.5 et 4Hz

Chaque stade de sommeil étant caractérisé par certains types d'onde ci-dessus, nous avons filtré le signal de chacun des 7 enregistrements EEG entre 0 et 30Hz et decoupé en quatre plages correspondant aux ondes citées. Sur chacune des plages, nous avons calculé la moyenne, la somme des amplitudes en valeur absolue ainsi que le ratio de la somme des amplitudes en valeur absolue de la la plage sur celle du signal total filtré, que l'on peut appeler amplitude relative. Nous calculons également la somme des amplitudes au carré du signal total ainsi que les proportions de chacune des plages de fréquences dans le signal.

Parallèlement à cette décomposition, nous réalisons une décomposition en quatre ondelettes des enregistrements EEG. Cette décomposition se fait après un filtre de Daubechies appliqué au signal. On obtient alors les coefficients de chacune des 4 ondelettes. Nous calculons ensuite l'écart-type et l'entropie de Renyi de chaque ondelette. L'entropie est une mesure de l'énergie du signal.

Après ces premiers calculs de features, nous remarquons que le stade de sommeil le plus dur à classer est le stade 1. Ce dernier est caractérisé notamment par les ondes alpha, c'est pourquoi nous décidons de calculer des features supplémentaires uniquement sur la plage de fréquence alpha des signaux EEG. Nous calculons ainsi l'écart-type, l'entropie de Renyi et la min-max distance de cette plage. La min-max distance est la somme des distances entre les points maximum et minimal du signal découpé en n intervalles.

4. Modèle utilisé - Description théorique

Nos tests sur les features nous ont permis de relever les plus influents et de sélectionner ceux de notre modèle final.

Nous utilisons l'algorithme de Random Forest pour la classification.

Théorie : voir notes du dernier cours (Gradient boosting)

5. Protocole de validation croisée

La validation croisée est réalisée par l'algorithme de Random Forest décrit plus haut.

6. Présentation des résultats

Choix du nombre d'arbres

Choix du nombre de variables tirées aléatoirement avec remise à chaque étape de l'algorithme

Matrice de transition finale sur le training set avec la validation croisée réalisée

7. Bibliographie